A Mobility-Supporting MAC Scheme for Bursty Traffic in IoT and WSNs

Georgios Z. Papadopoulos*, Vasileios Kotsiou†, Antoine Gallais†, George Oikonomou*,
Periklis Chatzimisios‡, Theo Tryfonas* and Thomas Noël†
*Faculty of Engineering, University of Bristol, UK *[g.papadopoulos, g.oikonomou, theo.tryfonas]*@bristol.ac.uk
†ICube Laboratory, University of Strasbourg, France *vkotsiou*@sch.gr, *[gallais, noel]*@unistra.fr
‡Alexander TEI of Thessaloniki and Hellenic Open University, Greece *peris*@it.teithe.gr

***Abstract*—Recent boom of mobile applications has become an essential class of mobile Internet of Things (IoT), whereby large amounts of sensed data are collected and shared by mobile sensing devices for observing phenomena such as traffic or the environmental. Currently, most of the proposed Medium Access Control (MAC) protocols mainly focus on static networks. However, mobile sensor nodes may pose many communication challenges during the design and development of a MAC protocol. These difficulties first require an efficient connection establishment between a mobile and static node, and then an efficient data packet transmissions. In this study, we propose MobIQ, an advanced mobility-handling MAC scheme for low-power MAC protocols, which achieves for efficient neighbour(hood) discovery and low-delay communication. Our thorough performance evaluation, conducted on top of Contiki OS, shows that MobIQ outperforms state-of-the-art solutions such as MoX-MAC, MOBINET and ME-ContikiMAC, in terms of significantly reducing delay, contention to the medium and energy consumption.**

***Index Terms*—Internet of Things, WSNs, Medium Access Control, Mobility, Neighbour Discovery, Bursty Traffic.**

I. Introduction

During the last years we experience the emergence of a new paradigm called the Internet of Things (IoT) in which smart, uniquely identifiable and connected objects that construct a network of things. Those things can communicate with each other or across the existing network infrastructure. In particular, the embedded and external wearable sensors constitute a major reason for the evolution of mobile devices. Indeed, they have drawn a lot of attention in healthcare monitoring applications providing more personalised services to users such as keeping track of health and well-being. In such applications, the requirements for instance mobility and support of bursty traffic very often appear to be crucial [1].

In mobility-aware environments, sensors are attached to humans, animals or objects, while the sensor nodes very often transmit the stored data in burst once they gain access to the wireless medium [2]. Medium Access Control (MAC) layer is in charge of the communications between nodes and handles all operations related to the main source of energy consumption, which is packet transmission and reception [3]. In IoT and Wireless Sensor Networks (WSNs), the MAC layer is also responsible for switching the radio device ON and OFF at periodic intervals. This duty-cycling functionality results in a fundamental trade-off between energy consumption and network performance.

Even though the number of mobile-based applications keeps growing, the recently formed IETF 6TiSCH working group focuses mainly on static networks [4]. Under these networks, the topology is considered fixed while the next-hop of each node may change mainly depending on physical layer conditions and device status (e.g., remaining battery, faults) [5]. Furthermore, a large number of MAC protocols have been proposed [6], [7], but to the best of our knowledge very few of them address the needs implied by the presence of mobile nodes and variable traffic in the network. As a result, no successful mobile IoT deployment with dynamic traffic has been experienced up to date.

In this paper, we propose and study an efficient mobility-supporting mechanism that achieves low-delay mobile-to-static communication. We place our investigation in the context of low power mobile nodes whose objectives are to dequeue their buffers once they detect a sink-connected neighbour within their vicinity. To this aim, we introduce a MAC scheme, namely MobIQ, which allows for selective and efficient neighbourhood discovery while it supports bursty and dynamic traffic. It also handles link fluctuations and disconnections that frequently occur due to node mobility. Furthermore, MobIQ mitigates the channel contention that frequently occurs due to the hidden terminal problem under heavy traffic loads. To this end, we focus on preamble-sampling Low-Power Listening (LPL) MAC protocols, mainly due to the topology and traffic dynamics induced by mobile nodes [6].

The contributions provided in this paper are threefold:

- After a thorough study of the MAC layer protocols for mobility-aware in the literature, we introduce MobIQ scheme that allows for selective and low-delay mobile to static communication. MobIQ is compliant with most of the preamble-sampling families of MAC protocols.
- We then introduce the Contention Avoidance Algorithm (CAA) for MobIQ to leverage channel contention and the hidden terminal problem. Hence, we show to what extent MobIQ allows reduced energy consumption, along with reduction of channel occupancy and attained delay.
- Finally, we perform a thorough performance evaluation campaign on top of COOJA (an emulator for Contiki OS). In addition, we compare MobIQ against state-of-the-art solutions such as ME-ContikiMAC [2], MoX-MAC [8] and MOBINET [9].

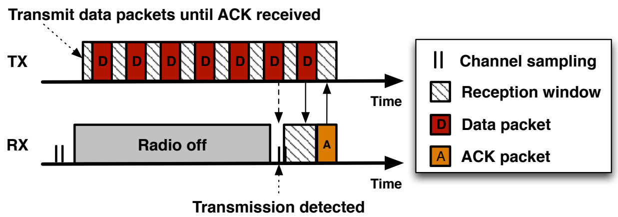


Fig. 1: A representative scheme from LPL MAC protocols. Nodes sample the medium periodically to detect a transmission. If a carrier is detected, the receiver keeps its radio ON to receive the associated data packet(s).

The remainder of our paper is organized as follows. In Section II we provide the necessary background information and we review the most pertinent related works from the literature. We present the foundations of the problem that we intend to address in Section III while Section IV focuses on the design of the proposed MobIQ. We then implement our solution on top of the Contiki OS (Section V) and demonstrate the attained performance of our proposed approach in Section VI. Finally, Section VII provides the concluding remarks as well as our future work.

II. Background & Related work

Considering the topology and traffic dynamics in a wireless network due to mobility, mobile nodes have to often perform neighbourhood discovery and handover procedures, since link (mobile to any static node) fluctuation and disconnection frequently occur [10]. To tackle the previously stated issues, we decided to integrate our proposed mechanism to the link layer, jointly with the LPL-based family of MAC protocols [6], [11]. Under LPL protocols nodes sample the medium at regular intervals to detect a carrier for incoming packets. A node expecting to transmit will have to repeatedly send the same data packet with a preamble period longer than the sampling frequency to ensure that its target is awakened, until a link layer acknowledgment is received (see Figure 1).

A number of related MAC layer schemes have been proposed in the literature and are summarized in [7]. Below we review the LPL-based key contributions that are mainly related to mobility-aware IoT and WSNs.

In [12], authors present the X-Machiavel scheme that allows mobile node to steal the wireless medium from a static node that has gained it earlier. Potential steals are detected by overhearing mobile nodes which prevent those nodes from low power operations, due to intensive sampling.

In [8], authors present the MoX-MAC protocol that, similarly to X-Machiavel, when a mobile node expects to transmit a data packet, it overhears the wireless medium to detect transmission between static nodes. It then waits until the end of the scheduled transmission and transmits its data packet to the static transmitter node. The efficiency of this approach strongly depends on the communication frequency between the static nodes.

MX-MAC [1] and MA-MAC [13] protocols are based on Received Signal Strength Indicator (RSSI). Under such solutions, if the mobile node evaluates a very low quality of the link between the current next-hop and itself, it initiates a neighbourhood discovery process which may lead to a handover necessity. However, under real-word scenarios the received signal level that is utilised as a mobility indication does not provide fair accuracy to evaluate proximity.

In [2], authors introduce ME-ContikiMAC, where mobile nodes expecting to transmit n data packets in burst, will transmit one additional control packet upfront in anycast. Thus, the first acknowledging node (of the control packet) will serve as temporary next-hop, and will forward the following data packets towards the border router. Note that the control packets are appropriately labeled in order not to be forwarded to avoid duplications in the network.

Under MOBINET [9], a mobile node, by overhearing the medium, builds a neighbourhood table with destination addresses of the static nodes within its transmission range. Later, when it expects to transmit, it unicasts a data packet to one of the destination addresses listed in its neighbourhood table. For the next-hop selection, MOBINET comes with two methods, the random and selective, respectively.

Most of the previously presented protocols may not efficiently satisfy our objectives of addressing mobility under bursty traffic in a highly proactive manner and by attaining low delay and energy consumption due to potentially longer routing paths. MoX-MAC, ME-ContikiMAC and MOBINET being proactive protocols, appear as the most relevant to our targeted context, and they are also independent from the underlying MAC protocol. Therefore, we selected these schemes as candidates for further comparison during our evaluation campaign.

III. MobIQ Overview

In this Section, we first discuss the design challenges related to mobility in IoT and WSNs as well as we present the selected underlying protocol for MobIQ, a mobile-supporting scheme that allows selective neighbour (hood) discovery.

A. Architecture Overview

In this paper, we assume that mobile nodes do not participate in the routing operation due to their velocity and sleep duration. Moreover, they may endanger the network connectivity or induce crippling communication costs to maintain a coherent routing backbone. Thus, we investigate a hybrid network architecture where a set of mobile nodes communicate with static low-power wireless sensor nodes.

We consider mobile nodes that aim at discovering neighbours in order to set up point-to-point communication links. We therefore place our work in the context of low power mobile nodes whose objectives are to dequeue their transmission buffers as soon as some sink-connected neighbours show up in their vicinity. Furthermore, we assume that the static nodes contain up-to-date routing layer information, i.e., RPL control packets. This information can be made available to the MAC layer of potentially surrounding mobile nodes, i.e., through ACK, in order to allow them to select the best next-hop without be aware of any information about the routing infrastructure.

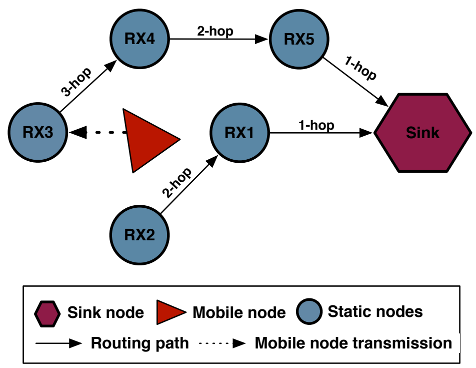


Fig. 2: A random next-hop selection (all RX1, RX2, RX3 and RX4 are neighbours of the mobile node).

B. Challenges of Random Selection

In most MAC protocols intended to handle mobility, mobile nodes aim at establishing communication links with randomly selected next-hop static nodes (e.g., ME-ContikiMAC). Such approaches can lead to increased end-to-end delay and energy consumption due to potentially longer routing paths [14].

Figure 2 illustrates those potential issues. Let us assume that a mobile node has n packets for transmission towards static infrastructure. It seems that within the transmission range of the mobile node, four static nodes exist, namely RX1, RX2, RX3 and RX4 that are 1, 2, 3, and 2 hops away from the sink, respectively. The mobile node, by utilising any of the previously presented random-based next-hop selection, may establish a link to any of those neighbouring static nodes (i.e., the first one that acknowledges its request). Thus, if receiver RX3, which is 3-hops away, samples the medium first and will respond to the request. It would then receive and forward the n data packets to the sink. Consequently, the end-to-end delay will increase and traffic in the network will also attain higher values, when compared to the RX1. As a result, by employing such opportunistic schemes, there is no guarantee that the mobile node will select its temporary next-hop based on certain qualitative criteria.

C. Selection of the Underlying Protocol

As previously mentioned, we here investigate the preamble-sampling MAC protocols, and thus, we employ Contiki-MAC [15] (the default MAC layer protocol in the commonly used Contiki OS) as the underlying protocol for our proposition. However, the core mechanism of MobIQ is general enough to be applied to any LPL-based MAC protocol.

ContikiMAC comes with a burst handling mechanism to anticipate high traffic load in the network [16]. To do so, the sender sets a flag at each data packet of the queue (except the last one) in order to notify the receiver that another packet will follow. On the other side, the receiver turns its radio ON and switches into Carrier Sense Multiple Access (CSMA) mode, during the burst period (see Figure 3). We here consider scenarios where mobile nodes transmit in burst, and thus, the burst notification flag of ContikiMAC is activated. However, like most of the LPL-based protocols, ContikiMAC does not obtain good performance under environments where static and mobile nodes co-exist. In fact, since mobile nodes do not participate in the routing construction, they are unable to transmit to static nodes in an efficient manner (i.e., by achieving low energy consumption and delay).

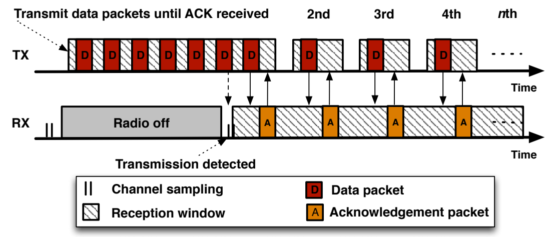


Fig. 3: ContikiMAC in burst mode: the receiver switches to CSMA mode to handle high traffic.

IV. Design of MobIQ

This Section presents MobIQ scheme that provides selective and efficient neighbourhood discovery, whereas in the same time supports dynamic and bursty traffic. We assume that mobile nodes, due to their nature, are not aware of the surrounding static sensors, as well as their distance (i.e., in terms of hops) from the sink. Thus, under the MobIQ scheme, mobile nodes exchange information (e.g., min-hop) with the static nodes during the neighbourhood discovery phase in order to achieve low end-to-end delay performance. This information exchange may adapt accordingly to the application layer requirements (for instance, the remaining energy to prioritize the relay node accordingly to the remaining energy level).

A. Neighbourhood Discovery

The MobIQ scheme consists of two concrete phases; (a) the neighbourhood discovery and (b) the transmission of n packets in a burst. The principles of the MobIQ scheme are depicted in Figure 4.

During the neighbourhood discovery phase, the mobile node (i.e., the transmitter TX) repeatedly transmits control packets in anycast transmission mode throughout the preamble period (e.g., $125\ ms$), in order to assure that all neighbours within its transmission range will successfully receive them. Note that the control packets are labeled not to be forwarded. In each control packet, the remaining time (i.e., $TXremaining$) of the "current" preamble period is included. On the other side, the potential receivers of the control packet (i.e., RX1, RX2, RX3), will respond with an acknowledgement that includes their unique identifier and an associated metric (e.g., link quality indicator, min-hop to the sink station, remaining battery power). Then, before switching their radio OFF, for energy saving purposes, they will calculate the remaining time to wake-up and receive the following n data packets.

In a later stage, based on the retrieved information from the acknowledgements, the mobile node identifies its temporary next-hop. Hence, during the second phase, the TX transmits the n packets in burst to the selected static node by achieving low end-to-end delay, by utilizing the $TXremaining$ information.

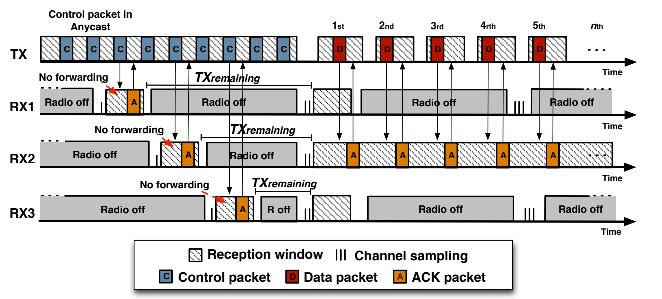


Fig. 4: MobIQ: after $TXremaining$ time, the potential receivers sample the medium to receive the data packets.

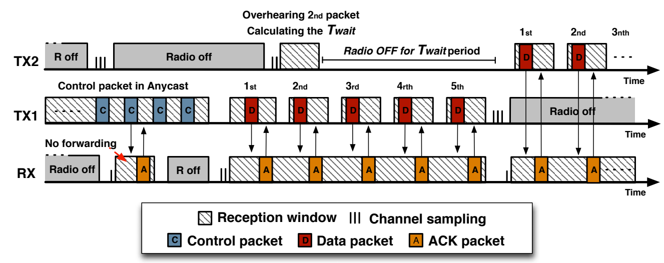


Fig. 5: Contention avoidance under CAA mode of MobIQ.

Note that MobIQ does not require high-quality clocks or costly time-synchronisation techniques (that are either not available in the hardware or too costly to be implemented). Moreover, the $TXremaining$ time does not require precise timing, since the transmitter nodes may wake-up slightly earlier to ensure successful reception.

B. Contention Avoidance Algorithm (CAA)

Inspired by the rendezvous time used in T-AAD [11], we here introduce the Contention Avoidance Algorithm (CAA) to mitigate the channel contention and hidden terminal problems. To this aim, we propose to incorporate the queue length information in every MobIQ's data packet. Thus, the surrounding medium access contenders, by overhearing this information, may estimate the total time of the burst transmission, and consequently they can calculate their sleep duration as follows:

$$T_{wait} = Q_{len}(TX_{time} + ACK_{time})(1 + M_{err}) + T_{ack} \quad (1)$$

where Q_{len} indicates the total number of packets that a a mobile node is expected to transmit (i.e., queue length) obtained from overhearing the data packet, TX_{time} and ACK_{time} are the time intervals for a successful exchange of data and ACK packets, and finally, in order to ensure that the calculated T_{wait} period is at least as long as the total burst transmission (including MAC retransmissions), a margin of error (i.e., M_{err}) is added. We consider M_{err} to cope with the potential packet retransmissions due to external interference, which can be determined according to the occurring network conditions [11].

However, due to the hidden terminal problem, we may observe interference and collisions during the burst transmission. Figure 5 shows an example case where two transmitters (i.e., TX1 and TX2) are sending data packets to a single receiver (i.e., RX). Let's assume that TX1 and TX2 have 5 and n data packets to transmit, respectively, while TX1 and TX2 are hidden to each other. To overcome this issue, we propose to consider a receiver node to announce the queue length information of the mobile transmitter in its acknowledgment. Thus, by overhearing the ACK, a hidden contender (i.e., TX2) will calculate the TX_{wait} as follows.

$$T_{wait} = Q_{len}(TX_{time} + ACK_{time})(1 + M_{err}) \quad (2)$$

Thus, the surrounding contenders (both mobile and static nodes) may postpone their data transmissions for a T_{wait} period by turning their radio OFF to save energy, while the transmitter mobile node will efficiently utilise the channel for a whole burst transmission period without being interfered.

V. Implementation Aspects

For our evaluation campaign, we have implemented MobIQ in the Contiki OS with Tmote-Sky platform (i.e., CC2420 radio chip at 2.4 GHz) and run a set of simulations over COOJA. We have utilised the BonnMotion tool to support mobility in the network. Furthermore, regarding the energy estimation, we employed the Contiki `energest` module to log the radio on-time. This module monitors in real-time the radio and Central Processing Unit (CPU) usage by saving the duration spent in each state.

Hereafter, we will detail the modifications performed on Contiki OS, in order to implement the MobIQ scheme.

Gradient: In this campaign, we used the RIME communication stack [17], while for the data collection scheme we rely on a low over-headed and scalable (under realistic conditions gradient) protocol that generates and maintains a tree-based routing topology, rooted at the sink.

Timers: We implemented both $TXremaining$ (reserves 2 bytes) and T_{wait} (Q_{len} reserves 1 byte) at the Radio Duty Cycle (RDC) layer for platform independent purposes. However, for better accuracy, they can be also implemented at the radio layer.

Software Acknowledgments: We disabled the hardware acknowledgments (i.e., "autoack") from the default functionality of Contiki OS, in order to implement the anycast transmission mode. By disabling hardware acknowledgments, the interrupt will be triggered at the MAC layer (i.e., radio driver).

CCA: By introducing the software acknowledgements, the time interval from the packet reception to its acknowledgement is increased. As a result, collisions between transmitter's consecutive data packet transmissions in burst and the receiver's acknowledgements may appear. To overcome this problem, we increased the time interval (i.e., t_i) between two consecutive data packets. Consecutively, in order to reliably detect a transmission, we increased from two to three the ContikiMAC's default inexpensive Clear Channel Assessment (CCA) checks, while the time interval between each CCA (i.e., t_c) remains unchanged (see Figure 6).

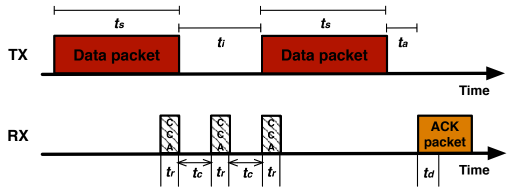


Fig. 6: A packet transmission and CCA timing under MobIQ scheme.

VI. Performance evaluation

Hereafter, we present a thorough performance evaluation of the MobIQ scheme. Even though MobIQ is generic-enough to employ various (routing) layer metrics (e.g., battery power, link quality, end-to-end delay, distance to the sink station), we here present results obtained from schemes utilising a min-hop metric. For comparison purposes, we implemented and compared MobIQ against state-of-the-art solutions such as MOBINET [9], MoX-MAC [8] and ME-ContikiMAC [2].

Our simulation environment involves 40 fixed nodes (including the sink) that are uniformly distributed in an area of $50 \times 40\ m$, with network degree 6.15 in average. In addition, we involved 8 mobile nodes that move within the area covered by the fixed nodes, by employing a random waypoint mobility model, with three different velocities. More specifically, the low speed (i.e., from $0.5\ m/s\ to\ 2\ m/s$) that represents a human walking, medium speed (i.e., from $2\ m/s\ to\ 8\ m/s$) that represents a typical jogging speed and high speed (i.e., from $8\ m/s\ to\ 12\ m/s$) that represents cycling speed. In this study, we present application-dependent (i.e., time-driven) results where mobile nodes employ a bulk transmission scheme of 32 packets every 120 sec while the static nodes transmit by utilising a Constant Bit Rate (CBR) 1 pkt per 30 sec, having as a result more than 13000 $pkts$ transmissions in total. We choose the packet size to be equal to 33 $bytes$ that corresponds to all necessary information for MAC, routing and application operations. Furthermore, we set our network to run with the Unit Disk Graph Medium (UDGM), where each node emits at $-15\ dBm$ transmission power, imposing multi-hop communication among the mobile nodes and the sink (up to 7 hops). The details of the simulation setup are exposed in Table I.

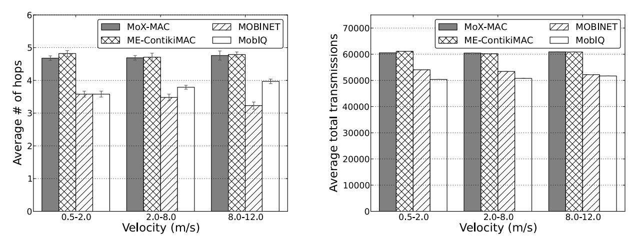
The results hereinafter show the performance of the studied schemes in terms of traffic, delay, energy consumption and reliability.

A. Traffic analysis

Under MobIQ with min-hop metric, mobile nodes select as temporary parent (i.e., next-hop) the static node that is closer (in terms of hops) to the border router among all the static nodes that are located within its vicinity. Indeed, as can be observed from Figure 7a, MobIQ chooses its next-hop node by at least 1-hop closer in average when compared to the randomly-based neighbour discovery schemes. By doing so, MobIQ achieves to reduce the total packet transmissions as well as the unnecessary transmissions in the network (see Figure 7b). As a result, MobIQ reduces the channel occupancy,

Topology parameters	**Value**
Topology	50×40
Nodes distribution	Grid
Number of nodes	40 fixed & 8 mobile sensors
Number of sources	47
Node spacing	$x = 6\ m\ /\ y = 8\ m$
Network degree	6.15
Mobility parameters	**Value**
Mobility model	Random waypoint
Velocity	Low speed: from 0.5 $m/s\ to$ 2 m/s Medium speed: from 2 $m/s\ to$ 8 m/s High speed: from 8 $m/s\ to$ 12 m/s
Simulation parameters	**Value**
Duration	68 $minutes$
Data collection scheme	Mobile nodes: Burst: 32 $pkts/120\ s$ Static nodes: CBR: 1 $pkt/30\ s$
Number of events	Mobile nodes: 8192 $pkts$ Static nodes: 5108 $pkts$
Payload size	38 $Bytes$
Routing model	Static network: Gradient Mobile nodes: Opportunistic
Number of hops	Multihop (7 hops maximum)
MAC model	Mobile nodes: MobIQ, MOBINET, MoX-MAC, ME-ContikiMAC Static nodes: ContikiMAC
Sampling frequency	125 ms, (with 3 maximum retries)
Hardware parameters	**Value**
Antenna model	Omnidirectional CC2420
Radio propagation	2.4 GHz
Modulation model	O-QPSK
Transmission power	$-15\ dBm$

TABLE I: Simulation setup.



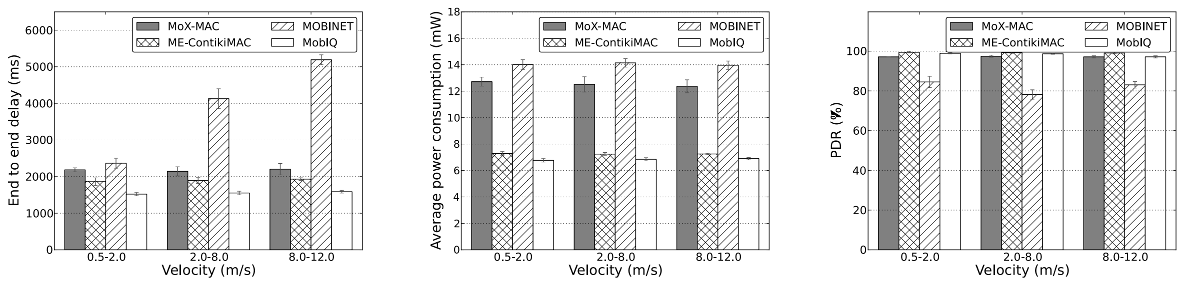
(a) Average number of hops (from mobile node to the sink).
(b) Total transmissions in the network.

Fig. 7: Traffic analysis.

the competition for medium access and the congestion which in turn decreases the probability of packet retransmissions (that have a major impact on delay performance) due to potential collisions.

B. Delay

Figure 8a illustrates the average end-to-end (from mobile to sink node) delay per data packet transmission. Both 1-hop and end-to-end delay include the channel sampling period, initial back-off, potential congestion back-off, potential retransmission delay and the transmission time of the preamble. MobIQ outperforms all three protocols and significantly improves the delay performance for all considered scenarios (i.e., velocities). These results are mainly due to the efficient next hop selection method that we have presented in the previous Section, which chooses as temporary next-hop the static node that is closer to the sink station in terms of hops.



(a) Average end to end delay (from mobile to sink).

(b) Average energy consumption per second for the whole network.

(c) Packet delivery ratio.

Fig. 8: Performance evaluation of MobIQ scheme in terms of end-to-end delay, energy consumption and reliability.

C. Energy consumption

As it can be observed from Figure 8b, MobIQ consumes less energy network-wide when compared against other schemes. In particular, it reduces energy consumption by up to the 6%, 45% and 51%, respectively. This enhanced performance is due to the reduction of the hop count towards the border router, and consequently, of the total number of packet transmissions in the network, (see Figure 7b). On the other hand, MoX-MAC and MOBINET increase the contention (and, thus, the retransmissions) to the medium access due to the presence of large number of collisions in the network. Finally, we should take into account the fact that schemes such as MoX-MAC and MOBINET are based on overhearing techniques, which actually means that the radio remains active for longer time.

D. Reliability

For each scheme, we calculate the Packet Delivery Ratio (PDR), in which packet loss is calculated as $1 - PDR$, and thus, packet loss 0% is the equivalent of 100% PDR. Our simulation results show that the optimization introduced by MobIQ does not impact the network reliability in a negative manner. Indeed, MobIQ achieves a very hight reliability performance (i.e., above 98%), similar to ME-ContikiMAC (see Figure 8c).

VII. Conclusions and future work

In this work, we have tackled the mobile-to-static communication and its potential congestion issue for low-power MAC protocols under bursty traffic. After a thorough study of the state of the art, we have introduced MobIQ, an efficient neighbour discovery scheme. We have also introduced a Contention Avoidance Algorithm (CAA) to mitigate the contention and congestion problem that arises from the hidden nodes. We then performed a thorough simulation campaign and have shown that MobIQ provides reliable, low delay and energy efficient communication between mobile and static nodes. Our ongoing and future work consists of further investigating innovative mobile-supporting scheme for mobile IoT. Moreover, our vision is to further assess MobIQ by performing a set of experimental studies over a large scale testbed such FIT IoT-LAB. Thus, we plan to evaluate our mechanism under real-world environment and optimise it by learning from the challenges that may arise from the experimental procedure.

References

[1] W. Dargie and J. Wen, "A Seamless Handover for WSN Using LMS Filter," in *Proceedings of the 39th IEEE Conference on Local Computer Networks (LCN)*, 2014, pp. 287–288.

[2] G. Z. Papadopoulos, V. Kotsiou, A. Gallais, P. Chatzimisios, and T. Noel, "Wireless Medium Access Control under Mobility and Bursty Traffic Assumptions in WSNs," *Springer Mobile Networks and Applications (MONET)*, vol. 20, no. 5, pp. 649–660, 2015.

[3] J. Eriksson, F. Österlind, N. Finne, A. Dunkels, N. Tsiftes, and T. Voigt, "Accurate network-scale power profiling for sensor network simulators," in *Proceedings of the 6th European Conference on Wireless Sensor Networks (EWSN)*, 2009, pp. 312–326.

[4] "IEEE Standard for Low-Rate Wireless Personal Area Networks (LR-WPANs)," IEEE Std 802.15.4-2015 (Revision of IEEE Std 802.15.4-2011), April 2016.

[5] T. Winter, P. Thubert, A. Brandt, J. Hui, R. Kelsey, P. Levis, K. Pister, R. Struik, J. Vasseur, and A. R., "RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks," RFC 6550, 2012.

[6] G. Z. Papadopoulos, "Improving Medium Access in Dynamic Wireless Sensor Networks," Ph.D. dissertation, University of Strasbourg, 2015.

[7] R. Silva, J. S. Silva, and F. Boavida, "Mobility in wireless sensor networks - Survey and proposal," *Computer Communications*, vol. 52, pp. 1 – 20, 2014.

[8] P. D. Ba, I. Niang, and B. Gueye, "An Optimized and Power Savings Protocol for Mobility Energy-aware in Wireless Sensor Networks," *Telecommunications Systems*, vol. 55, no. 2, pp. 271–280, 2014.

[9] D. Roth, J. Montavont, and T. Noel, "MOBINET: Mobility management across different wireless sensor networks," in *IEEE Wireless Communications and Networking Conference (WCNC)*, 2011, pp. 351–356.

[10] G. Z. Papadopoulos, V. Kotsiou, A. Gallais, P. Chatzimisios, and T. Noel, "Optimizing the Handover Delay in Mobile WSNs," in *Proceedings of the 2nd IEEE World Forum on Internet of Things (WF-IoT)*, 2015, pp. 210–215.

[11] G. Z. Papadopoulos, J. Beaudaux, A. Gallais, and T. Noel, "T-AAD: Lightweight Traffic Auto-ADaptations for Low-power MAC Protocols," in *Proceedings of the 13th IEEE IFIP Annual Mediterranean Ad Hoc Networking Workshop (MED-HOC-NET)*, 2014, pp. 79–86.

[12] R. Kuntz, J. Montavont, and T. Noël, "Improving the medium access in highly mobile Wireless Sensor Networks," *Telecommunication Systems*, vol. 52, no. 4, pp. 2437–2458, 2013.

[13] T. Zhiyong and W. Dargie, "A Mobility-Aware Medium Access Control Protocol for Wireless Sensor Networks," in *Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM) Workshops*, 2010, pp. 109–114.

[14] G. Z. Papadopoulos, V. Kotsiou, A. Gallais, P. Chatzimisios, and T. Noel, "Low-Power Neighbor Discovery for Mobility-Aware Wireless Sensor Networks," *Elsevier Ad Hoc Networks*, vol. 48, pp. 66–79, 2016.

[15] A. Dunkels, "The ContikiMAC Radio Duty Cycling Protocol," *Swedish Institute of Computer Science, Technical Report*, 2011.

[16] S. Duquennoy, F. Österlind, and A. Dunkels, "Lossy Links, Low Power, High Throughput," in *Proceedings of the 9th ACM Conference on Embedded Networked Sensor Systems (Sensys)*, 2011, pp. 12–25.

[17] A. Dunkels, F. Österlind, and Z. He, "An Adaptive Communication Architecture for Wireless Sensor Networks," in *Proc. of the 5th ACM Conference on Embedded Networked Sensor Systems (SenSys)*, 2007.